

Vysoká škola báňská — Technická univerzita Ostrava

Fakulta strojní

Katedra automatizační techniky a řízení

Vývoj aplikací pro platformu Raspberry Pi

Development of Application for Raspberry Pi Platform

Student:

Bc. David Grygar

Vedoucí diplomové práce:

doc. Ing. Marek Babiuch, Ph.D.

Ostrava 2017

Zadání diplomové práce

Student: **Bc. David Grygar**
Studijní program: N2301 Strojní inženýrství
Studijní obor: 3902T004 Automatické řízení a inženýrská informatika
Téma: **Vývoj aplikací pro platformu Raspberry Pi**
Development of Application for Raspberry Pi Platform
Jazyk vypracování: čeština

Zásady pro vypracování:

1. Popište Platformu Raspberry Pi, její vývojové verze, hardwarové parametry a možnosti softwarového vývoje.
2. Navrhněte aplikaci na platformě Raspberry pro sběr a zpracování naměřených dat s využitím bezdrátové komunikace, webového serveru a s možností vzdáleného ovládání zařízení.
3. Rozšiřte navržené řešení o možnost komunikace zařízení Raspberry Pi se smart zařízením s OS Android.
4. Zhodnoťte dosažené výsledky a navrhněte směr dalšího řešení.

Seznam doporučené odborné literatury:

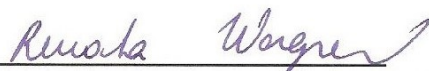
ALLEN, Grant. Android 4: průvodce programováním mobilních aplikací. 1. vyd. Překlad Jakub Mužík. Brno: Computer Press, 2013, 656 s. ISBN 9788025137826.
Android Developer portal . Dostupné z: <https://developer.android.com/index.html>
Raspberry Pi Official web sites. Dostupné z: <https://www.raspberrypi.org/resources/learn/>
Kivy Pie Kivy onRaspberry PI - Python framework for multitouch apps. Dostupné z: <http://kivypie.mitako.eu/kivy-pie.html>

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

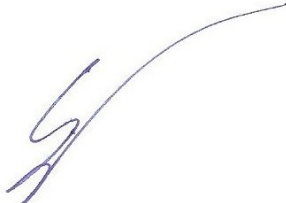
Vedoucí diplomové práce: **doc. Ing. Marek Babiuch, Ph.D.**

Datum zadání: 09.12.2016

Datum odevzdání: 15.05.2017


doc. Ing. Renata Wagnerová, Ph.D.
vedoucí katedry




doc. Ing. Ivo Hlavatý, Ph.D.
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě: 15. května 2017

.....

podpis studenta

Prohlašuji, že

- jsem byl seznámen s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě diplomovou práci užit (§ 35 odst. 3).
- souhlasím s tím, že diplomová práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užit dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užit své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- beru na vědomí, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě: 15. května 2017

.....
podpis

Jméno a příjmení autora práce: David Grygar

Adresa trvalého pobytu autora práce: Kyjovice 178, 747 68 Kyjovice

Poděkování

Chtěl bych tímto poděkovat vedoucímu své diplomové práce za vedení rady a pomoc při vypracovávání a zároveň děkuji své rodině a přítelkyni, kteří mi poskytovali po celou dobu morální podporu.

Anotace

Grygar, D. *Vývoj aplikací pro platformu Raspberry Pi: diplomová práce*. Ostrava: VŠB – Technická univerzita Ostrava, Fakulta strojní, Katedra automatizační techniky a řízení, 2017, 48 s. Vedoucí práce: Babiuch, M.

Tato diplomová práce se zabývá mikropočítačem Raspberry Pi. Práce obsahuje informace o historii platformy, technických parametrech a předchozích verzích jednodeskového počítače. Hlavní část popisuje realizaci mobilní a webové aplikace pro vizualizaci a archivaci měřených dat ze senzorů a také možnost vzdáleného ovládání zařízení. Pro tyto aplikace jsou využity programovací jazyky PHP, Java, databázový systém MySQL a další. Přenos dat je řešen bezdrátově s pomocí technologie Bluetooth a Wi-Fi a data jsou posílána mimo jiné ve formátu JSON. Práce si klade za cíl možnost reálného použití aplikací v moderních domácnostech.

Annotation

Grygar, D. *Development of Application for Raspberry Pi Platform: Master Thesis*. Ostrava: VŠB – Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Control Systems and Instrumentation, 2017, 48 p. Thesis head: Babiuch, M.

The master thesis deals with Raspberry Pi microcomputer. Thesis contain information about platform, history, technical parameters and previous version of single-board computer. Main part of thesis describe design and development of application for this microcomputer with the use of programming language Python. Further describe realization of mobile and web application for visualization and archiving measured data from sensors and also remote control of devices. For this applications are used programming language PHP, Java, database system MySQL and more. Data transmission is solved by wireless communication using Bluetooth and Wi-fi technology. Also is used JSON format for data transmission. The goal of thesis is possibility of real use in modern homes.

Obsah

Seznam použitých značek a symbolů.....	8
1 Úvod	11
2 Charakteristika platformy Raspberry Pi	12
2.1 Historie platformy	13
2.2 Vývoj a parametry RPI	13
2.3 Softwarová podpora	18
3 Aplikace na Raspberry Pi	19
3.1 Měření teploty a tlaku	19
3.2 Měření intenzity osvětlení.....	21
3.3 Detekce uzavření/otevření dveří.....	23
3.4 Kamerové snímkování.....	25
3.5 Ovládání světel	26
3.6 Aplikace pro dotykový displej	27
4 Webová aplikace.....	28
4.1 Ovládání světel	30
4.2 Vizualizace.....	31
5 Mobilní aplikace	39
6 Závěr.....	42
Seznam použité literatury	44
Seznam použitých obrázků	47

Seznam použitých značek a symbolů

Ω – Ohm, jednotka elektrického odporu

$^{\circ}\text{C}$ – Celsiův stupeň, jednotka teploty

A – ampér, jednotka elektrického proudu

ARM – Advanced RISC Machine, označení procesorů

API – Application Programming Interface, rozhraní pro programování aplikací

C# – programovací jazyk

C++ – programovací jazyk

CSS – Cascading Style Sheets, kaskádové styly

GB – Gigabyte, jednotka digitální informace

GHz – Gigahertz, jednotka frekvence

GND – Ground, uzemění

GPIO – General-purpose input/output, Vstupy / výstupy pro všeobecné účely

HDMI – High-Definition Multi-media Interface, zvuková a obrazový signál

hPa – Hektopascal, jednotka tlaku

HTML – HyperText Markup Language, značkovací jazyk pro hypertext

I – elektrický proud

I2C – Inter-Integrated Circuit, sběrnice sloužící pro komunikaci mezi integrovanými obvody

IDLE – Integrated Development Environment

IoT – Internet of Things, Internet věcí

JSON – JavaScript Object Notation, JavaScriptový objektový zápis

kHz – kilohertz, jednotka frekvence

LED – Light-Emitting Diode, dioda emitující světlo

lx – lux, jednotka intenzity osvětlení

m – metr, jednotka vzdálenosti

MB – Megabyte, jednotka digitální informace

Mbps – megabit za sekundu

MHz – Megahertz, jednotka frekvence

MMC – MultiMediaCard, paměťová karta

OS – Operating System, operační systém

P – elektrický výkon

PC – personal computer, osobní počítač

PHP – Hypertextový procesor, Personal Home Page (původní název)

RAM – Random-access memory, polovodičová paměť

RCA – Radio Corporation of America, audio, video konektor

RJ45 – konektor síťových kabelů

RPI – Raspberry Pi

s – sekunda, jednotka času

SCL – Synchronous Clock, hodinový signál

SD – Secure Digital, paměťová karta

SDA – Synchronous Data, datový kanál

SMBus – System Management Bus, dvouvodičová sběrnice

SQL – Structured Query Language, strukturovaný dotazovací jazyk

TCP/IP – Transmission Control Protocol/Internet Protocol

U – elektrické napětí

URL – Uniform Resource Locator, jednotná adresa zdroje

USB – Universal Serial Bus, univerzální sériová sběrnice

V – Volt, jednotka elektrického napětí

WiFi – standardy popisující bezdrátovou komunikaci

XML – Extensible Markup Language, rozšiřitelný značkový jazyk

1 Úvod

Tato práce bude pojednávat o vývoji aplikací na platformě Raspberry Pi 3. Raspberry Pi je mikropočítač, který bude sloužit ke sběru dat, k realizaci akčních zásahů, pro bezdrátové odesílání dat a k přijímání příkazů. To bude zajišťovat skript, napsaný v jazyku Python. Současně byla vytvořena aplikace pro dotykový displej s použitím Kivy frameworku.

Součástí práce je také popis vývoje webové aplikace, která s mikropočítačem komunikuje pomocí technologie Wi-Fi a je dostupná z globální sítě Internet. Úkolem tohoto webového serveru je příjem, archivace, vizualizace a odesílání dat. Serverová část je naprogramována v jazyce PHP. Aplikace využívá zásady objektově orientovaného programování.

Vizualizace je vytvořena pomocí jazyků HTML, CSS, JavaScript. Data jsou uchována v relační databázi MySQL. Webová aplikace pracuje se všemi získanými daty z mikropočítače, mezi které patří: teplota, tlak, světelná intenzita, fotografie a údaje z magnetického senzoru. Rovněž vysílá data pro dvouhodnotové ovládání zařízení, konkrétně světel v domácnosti. Vizualizace obsahuje aktuální hodnoty s přihlédnutím na vzorkovací periodu, historické grafické trendy, tabulkové záznamy a extrémní hodnoty.

V rámci práce byla také vyvinuta mobilní aplikace pro telefony s operačním systémem Android, pro komunikaci využívající technologii Bluetooth. Aplikace obsahuje aktuální hodnoty meteorologických dat a dvouhodnotové ovládání. Jedná se o nativní aplikaci, která se instaluje do zařízení. Její výhodou je možnost funkčnosti i bez připojení ke globální síti Internet. Při programování aplikací pro operační systém Android byl využit programovací jazyk Java.

Práce si klade za cíl vytvořit nástroje pro použití v moderních domácnostech a cílí na problematiku internetu věcí. Vývoj byl soustředěn především na softwarovou část.

2 Charakteristika platformy Raspberry Pi

Pod pojmem Raspberry Pi si nesmíme představit pouze jeden produkt, ale sérii produktů, které prošly určitým vývojem. Hlavní definice však zní, že se jedná o mikropočítač za zajímavou nákupní cenu, což je i důvod jeho rozšíření. Pro ilustraci aktuální nejnovější model Raspberry Pi 3 Model B lze v tomto období (duben-květen 2017) koupit v české distribuci od 1 100 Kč.



Obrázek 1 – Raspberry Pi 3 Model B (HALFACREE 2017)

Další část, kterou nesmíme opomenout, jsou přídatné moduly, které mohou plnit specifické funkce. Je důležité mít na paměti, že pro provoz samotného mikropočítače budeme ideálně potřebovat tyto periférie: monitor/displej s HDMI vstupem, HDMI kabel, počítačovou USB myš, USB klávesnici. Pokud chceme použít naše programy tak, že budou interagovat s okolním reálným světem, musíme mít k dispozici příslušné moduly a senzory, které nám budou řešit řízení a realizaci akčních zásahů nebo obstarávat měření v rámci zpětné vazby.

Za zmínku také stojí, že programování samotného mikropočítače můžeme provádět přímo v něm a nemusíme tím pádem nahrávat externě vytvořené programy. Pro tyto účely musíme mít nainstalovaný OS a příslušné nástroje. (SLEDGE 2011), (THE RASPBERRY PI FOUNDATION 2016)

2.1 Historie platformy

První idea na vznik cenově dostupného počítače pro děti vznikla v roce 2006 na University of Cambridge. Autorům nápadu šlo o to, aby se děti mohli učit programovat, aniž by museli jejich rodiče vynakládat vyšší sumy na nákup počítače. Dnes však již Raspberry není jen počítač pro děti, ale nachází mnohem širší uplatnění. The Raspberry Pi Foundation, tak zní název organizace, která za celým projektem stojí. Tato organizace však není komerční firmou, ale má status charity, konkrétně vzdělávací charity.

Od roku 2008 se procesory vytvářené pro mobilní telefony začaly parametricky zlepšovat a to umožnilo budoucí vznik Raspberry Pi Model B o tři roky později. Přehled jeho parametrů, stejně jako i parametrů dalších modelů naleznete níže. Následoval vývoj dalších modelů, ale také softwarové podpory a přídatných modulů. (THE RASPBERRY PI FOUNDATION 2016)

2.2 Vývoj a parametry RPI

Pomineme-li různé prvotní prototypy, můžeme při popisu vývoje začít od Raspberry Pi 1 Model B. V tomto přehledu jsou k vidění chronologicky seřazené hlavní modely Raspberry Pi. U každého modelu jsou uvedeny základní specifikace.

Raspberry Pi 1 Model B (2012)

Procesor: BCM283

Frekvence procesoru: 700 MHz

Operační paměť (RAM): 256 MB, později 512 MB

Počet USB portů: 2krát – USB 2.0

Ethernet port: 10/100 konektor RJ45

Video výstupy: HDMI, RCA / composite

Audio výstup: HDMI; 3,5 mm jack

Čtečka karet: SD, MMC

GPIO: 26

Raspberry Pi 1 Model A

Procesor: BCM283

Frekvence procesoru: 700 MHz

Operační paměť (RAM): 256 MB

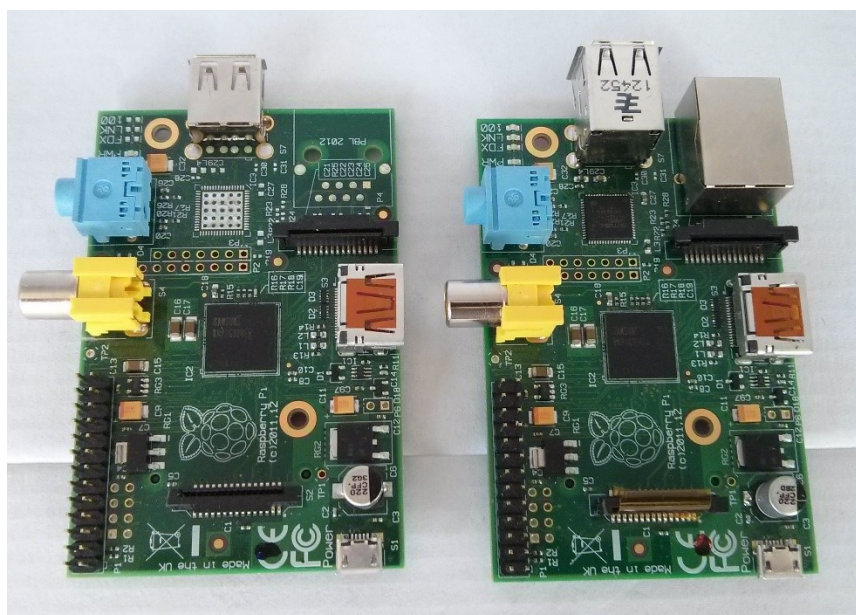
Počet USB portů: 1krát – USB 2.0

Video výstupy: HDMI, RCA / composite

Audio výstup: HDMI; 3,5 mm jack

Čtečka karet: SD, MMC

GPIO: 26



Obrázek 2 – Porovnání RPI Model A (vlevo) s RPI Model B (vpravo) (YOSHIDA 2013)

Raspberry Pi 1 Model B+

Procesor: BCM2835

Frekvence procesoru: 700 MHz

Operační paměť (RAM): 512 MB

Počet USB portů: 4krát – USB 2.0

Ethernet port: 10/100 konektor RJ45

Video výstupy: HDMI; 3,5 mm jack

Audio výstupy: HDMI; 3,5 mm jack

Čtečka karet: micro SD

GPIO: 40

Raspberry Pi 1 Model A+

Procesor: BCM2835

Frekvence procesoru: 700 MHz

Operační paměť (RAM): 256 MB

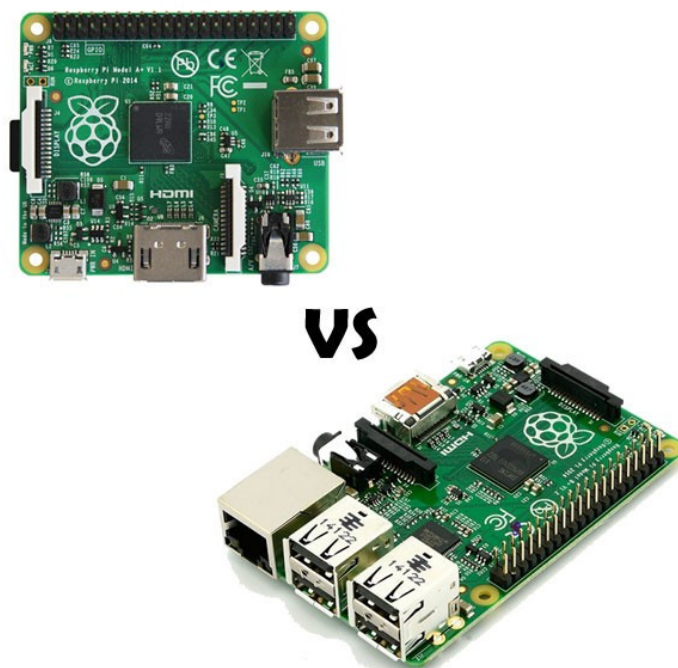
Počet USB portů: 1krát – USB 2.0

Video výstupy: HDMI; 3,5 mm jack

Audio výstupy: HDMI; 3,5 mm jack

Čtečka karet: micro SD

GPIO: 40



Obrázek 3 – Porovnání RPI model A+ (nahore) a model B+ (dole) (THE PI HUT 2017)

Raspberry Pi 2 Model B

Procesor: BCM2836 32 bit

Frekvence procesoru: 900 MHz

Operační paměť (RAM): 1 GB

Počet USB portů: 4krát – USB 2.0

Ethernet port: 10/100 konektor RJ45

Video výstupy: HDMI; 3,5 mm jack

Audio výstupy: HDMI; 3,5 mm jack

Čtečka karet: micro SD

GPIO: 40

Raspberry Pi 3 Model B

Jedná se o model použitý při zpracování této práce.

Procesor: ARM Cortex A53 64 bit

Frekvence procesoru: 1,2 GHz

Operační paměť (RAM): 1 GB

Počet USB portů: 4krát – USB 2.0

Ethernet port: 10/100 konektor RJ45

Video výstupy: HDMI; 3,5 mm jack

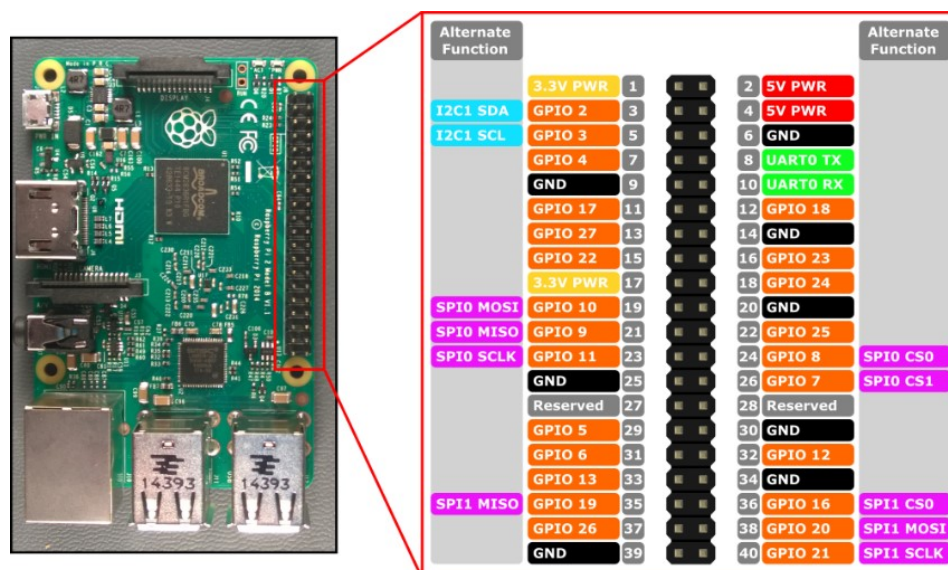
Audio výstupy: HDMI; 3,5 mm jack

Čtečka karet: micro SD

GPIO: 40

Wifi: ano, integrované (802.11n WiFi)

Bluetooth: ano, integrované (Bluetooth 4.1)



Obrázek 4 – Přehled pinů Raspberry Pi 3 Model B (MICROSOFT 2017)

Stručný přehled

V první modelové řadě byly 2 větve A a B, větev A měla pouze 1 USB port a neměla Ethernet port v porovnání s celou B větví. Poté zde byly 2 generace A, B a A+, B+. První neplusová generace měla pouze 26 GPIO a měla ještě RCA video vstup. Plusová (A+, B+) generace už měla 40 GPIO, což zatím platí pro všechny další RPI a dále přinesla micro SD slot, oproti předchozímu SD slotu na paměťové kartě. Následně už je vývoj přehlednější, protože existuje pouze jeden model druhé řady (RPI 2 Model B). Ten přináší rychlejší procesor. A nejnovější model RPI 3 Model B obsahuje již integrované Bluetooth a WiFi technologie, takže k práci již nepotřebujeme pro tyto technologie externí moduly. Od RPI 1 Model B+ si následující verze drží podobné rozmístění komponentů na desce a jejich vzhled je tudíž velmi podobný. (BENCHOFF, B. 2016) (ENGADGET 2016) (GE TECH WIKI 2014) (MICROSOFT 2016) (ROOT.CZ 2016)

2.3 Softwarová podpora

Oficiální podporovaný operační systém pro mikropočítače RPI je Raspbian. Raspbian obsahuje hodně předinstalovaného obsahu a může sloužit jako plnohodnotný OS pro práci s mikropočítačem. Zároveň je zde možné přímo vyvíjet programové kódy bez nutnosti dalšího zařízení. Tento systém byl použit pro tuto práci. V současné době (květen 2017) je nejnovější verze 4.4 s velikostí asi 4 GB. Systém obsahuje předinstalované prostředí pro vývoj v Pythonu, Javě nebo třeba .NET.

Dalším z možných operačních systémů je Windows 10 IoT, který neobsahuje téměř nic a je nutné aplikace externě vytvořit a následně teprve spustit. V současnosti je nejvhodnější prostředí pro vývoj programů Visual Studio 2015 a je vhodné ho mít nainstalované na PC s operačním systémem Windows 10. Jako další operační systémy pro RPI mohou být zmíněny tyto: Ubuntu MATE, OSMC, LibreELEC, PiNet, RISC OS. (JONES, A. 2015) (THE RASPBERRY PI FOUNDATION 2016)

Pro platformu Raspberry Pi existuje také velké množství knihoven, které jsou napsány především v jazyku Python, například pro senzory, komunikaci, kameru atd.

3 Aplikace na Raspberry Pi

Aplikace byly vyvíjeny přímo na mikropočítači v operačním systému Raspbian. Jsou napsány v programovacím jazyce Python. Python je vyvíjen jako open-source. Jedná se o dynamicky interpretovaný jazyk, což znamená, že se kód překládá až při vykonávání programu. Případné chyby se tedy vyskytnou až po jeho spuštění.

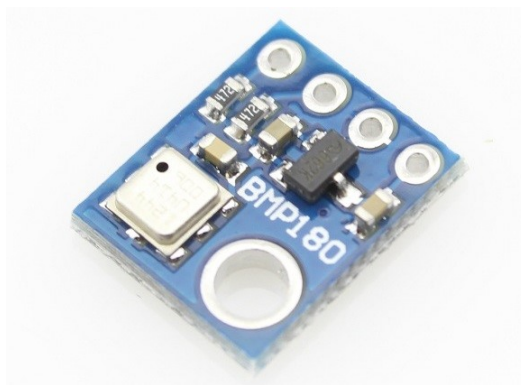
Pro strukturování kódu se nepoužívají závorky, ale odsazení tabulátorem. Podporuje procedurální, funkcionální i objektově orientované programování. Mezi další výhodu patří možnost pracovat s komplexními čísly. (GXC11 2014)

Vývoj aplikací nejprve probíhal postupně, kdy byla testována každá zvlášť. Hlavním cílem však bylo integrovat všechny aplikace do jednoho programu, což je vzhledem k reálnému využití nutné. Aplikace jsou navrženy pro získávání dat, ale i pro ovládání přístrojů. Mezi aplikace patří měření teploty, tlaku, světelné intenzity, detekce otevření pomocí magnetického senzoru, periodické snímkování, senzorové snímkování a aplikace pro ovládání dvoustavových spotřebičů.

Mikropočítač Raspberry Pi slouží pouze k záznamu, nikoliv k jejich archivaci. Ta je prováděna na webovém serveru. Aplikace mají informativní charakter a neslouží k zabezpečení domácnosti.

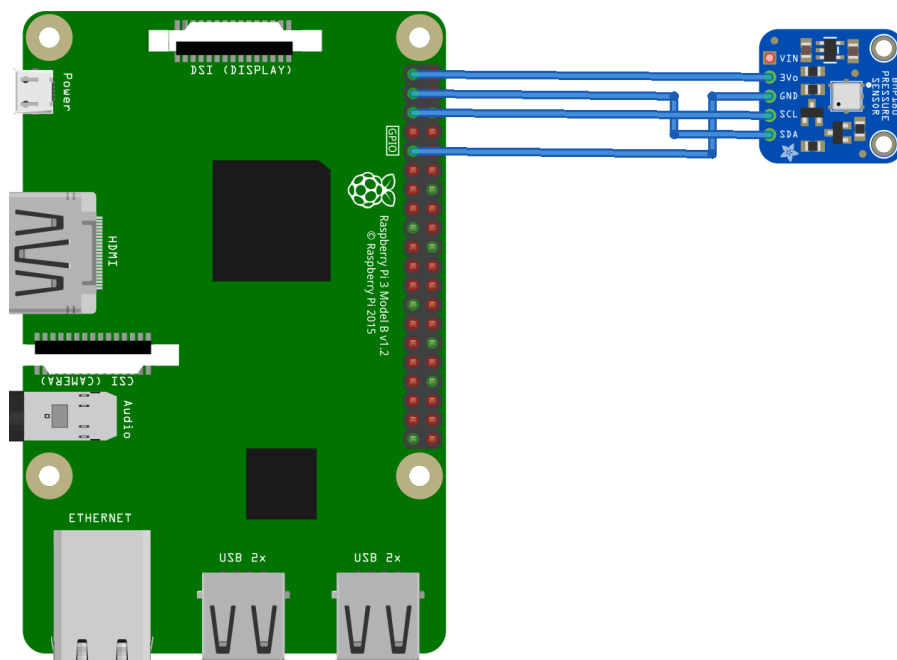
3.1 Měření teploty a tlaku

První z měřených meteorologických veličin je teplota, která je měřena pomocí senzoru BMP180 od firmy Bosch.



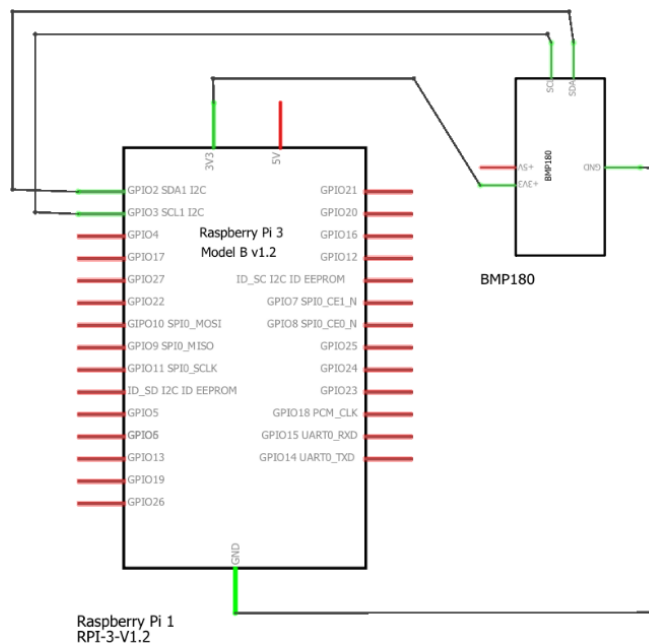
Obrázek 5 – Senzor BMP180 (EMBEDDED LAB 2015)

Tento senzor měří zároveň i tlak. Rozsah měřených teplot je od $-40\text{ }^{\circ}\text{C}$ do $+85\text{ }^{\circ}\text{C}$ a tlak od 300 do 1 100 hPa. Přesnost měření teploty je $0,1\text{ }^{\circ}\text{C}$ a tlak je měřen na celé jednotky pascalů. Senzor je z Raspberry Pi napájen pomocí 3,3 V pinu a musí být připojen na GND pin, což je uzemnění. Další dva výstupy realizují I2C komunikaci, pro kterou na Raspberry slouží piny 3 (SDA) a 5 (SCL). (BOSCH 2013)



fritzing

Obrázek 6 – Grafické schéma zapojení senzoru BMP180



fritzing

Obrázek 7 – Elektrické schéma zapojení senzoru BMP180

I2C (Internal-Integrated-Circuit Bus)

Jedná se o dvojdrátovou interní datovou sběrnici, která slouží pro přenos mezi integrovanými obvody, často v rámci jednoho zařízení. Přenos je realizován pomocí dvou vodičů. SDA – serial data a SCL – serial clock. Na jednu sběrnici může být připojeno více senzorů, protože každý má svou adresu. Raspberry Pi funguje jako tzv. master a senzory jako tzv. slave.

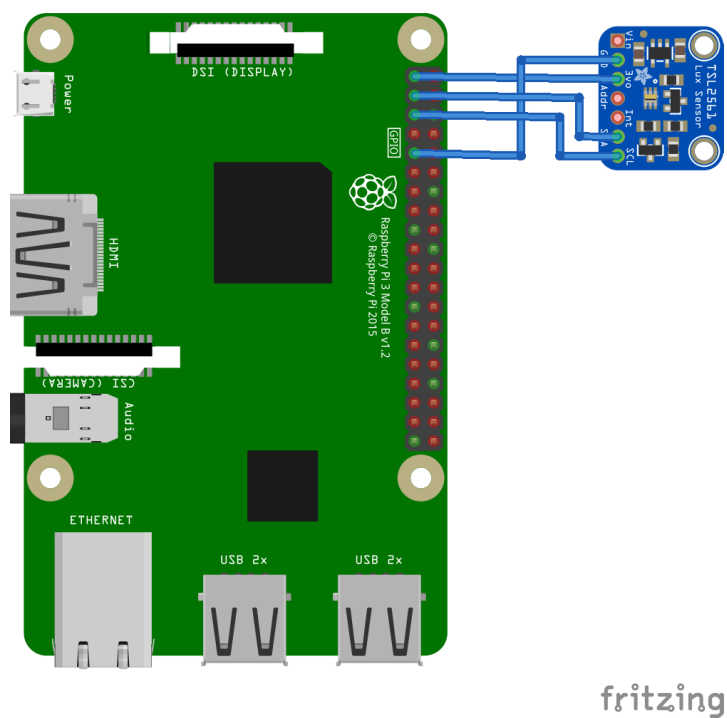
Měření teploty i tlaku je prováděno s periodou 60 sekund pro odesílání na webový portál a s periodou 5 sekund pro mobilní aplikaci. Vlákno pro měření a odesílání teploty a dalších meteorologických veličin je spuštěno pouze v dané časové okamžiky odpovídající vzorkovací periodě. Měření není prováděno neustále. Díky tomu je skript méně náročný. Zpracování dat webovým portálem bude dále zpracováno ve 4. kapitole. Respektive v 5. kapitole pro mobilní aplikaci.

3.2 Měření intenzity osvětlení

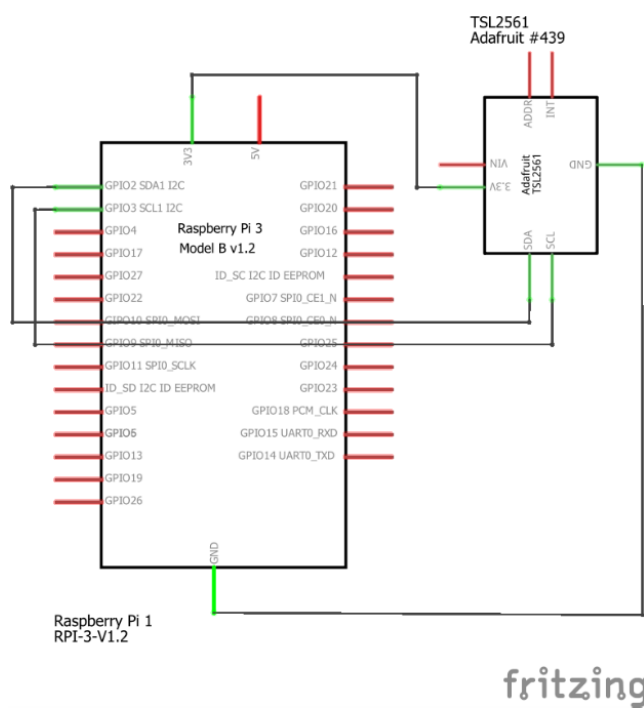
Měření intenzity osvětlení je realizováno pomocí senzoru TSL2561 od firmy TAOS, jehož rozpětí je od 1 do 40 000 luxů s přesností na celé jednotky. Jako rozhraní pro komunikaci může sloužit I2C nebo SMBus. SMBus je sběrnice odvozena z I2C a jsou si velmi podobné a při frekvenci do 100 kHz jsou i kompatibilní. Vzorkovací perioda je stejná jako u teploty a tlaku, stejně jako princip odesílání.



Obrázek 8 – Senzor TSL2561 (MG SUPER LABS 2017)



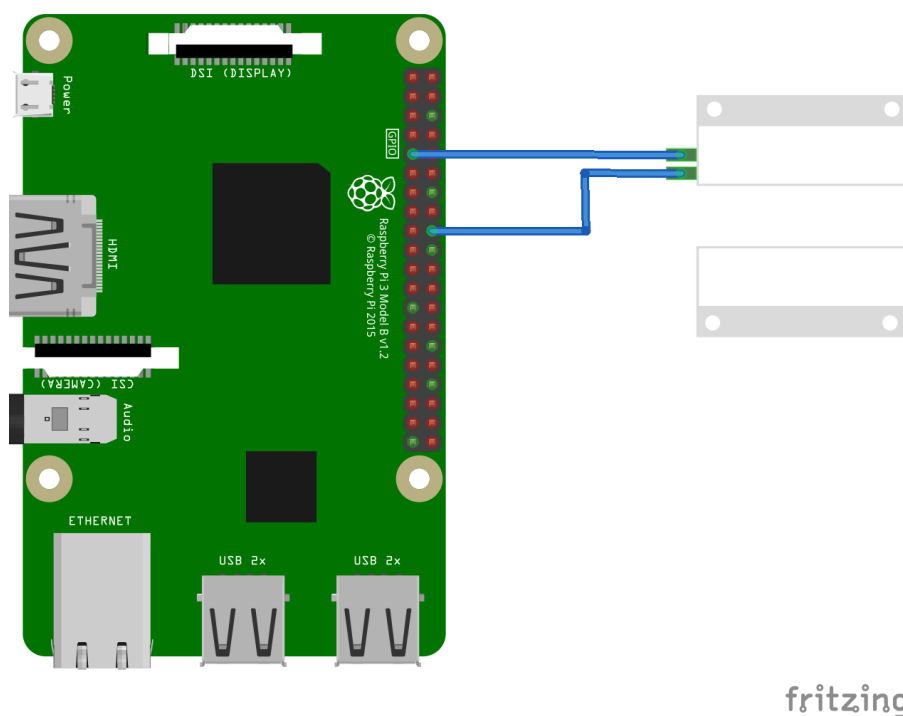
Obrázek 9 – Grafické schéma zapojení senzoru TSL2561



Obrázek 10 - Elektrické schéma zapojení senzoru TSL2561

3.3 Detekce uzavření/otevření dveří

Cílem aplikace je detekce otevřenosti/zavřenosti dveří, oken, brán, zásuvek nebo skříní. Základem měření je magnetický dvou vodičový senzor pro UniPi. Jeden z jeho vodičů je připojen na GND pin, tedy zem a druhý na jeden z mnoha pro toto použitelných pinů.



Obrázek 11 – Grafické schéma zapojení magnetického senzoru

V samotném kódu je nutné tento pin nastavit jako vstup, protože z něj chceme přijímat data a dále programově vytvořit pull up rezistor. Pull up rezistor slouží k určení logické hodnoty v případě, že není definována zařízením. Raspberry Pi má tyto rezistory zabudované na desce, tudíž nemusíme provádět fyzické zapojení rezistoru, ale jen programové. Nastavení pinu: `GPIO.setup(18, GPIO.IN, GPIO.PUD_UP)`. (HORNÍK 2015)

Pro naše potřeby, tedy rozpoznat, jestli jsou dveře otevřené nebo zavřené, nám stačí sledovat pouze nástupnou nebo sestupnou hranu signálu. Po změně signálu můžeme o systému prohlásit, že se nachází ve stejném stavu, nedošlo-li k další změně. Toto je možné tvrdit, protože jsme použili dříve zmíněný pull up rezistor. K detekci hran na jednotlivých pinech nám slouží metoda `add_event_detect()` z knihovny GPIO, která umožní detekci na daném pinu. Parametry metody můžeme nastavit pro detekci například jen nástupných hran, jen sestupných hran nebo všech hran, jako jsme to použili v naší

aplikaci. Konkrétní ukázka příkazu pro náš případ: `GPIO.add_event_detect(18, GPIO.BOTH)`.

Metoda `GPIO.event_detected()` je umístěna v nekonečné smyčce v podmínce, která když je splněna, spustí vlákno, které data odešle a spustí senzorové snímkování kamerou. Vzhledem k tomu, že sledujeme obě hrany, nemůžeme určit, jestli jsou dveře otevřené nebo zavřené. To lze provést příkazem `GPIO.input(18)`, který navrací hodnotu *true* (pravda) v případě zavřených dveří, tedy že jsou magnetické kontakty v blízkosti, nebo *false* (nepravda) v opačném případě.

```
if(GPIO.event_detected(18)):
    if GPIO.input(18): stav_door=1
    else: stav_door=2

    if(stav_door!=minuly_stav):
        threadMagnetic = Thread(target = magnetic_sensor,
        args = (stav_door,))
        threadMagnetic.setDaemon(True)
        threadMagnetic.start()

    if GPIO.input(18): minuly_stav=1
    else: minuly_stav=2
```

Všechny výše uvedené aplikace používají pro odesílání dat metody POST, která skrývá odesílaná data v těle. Dotazuje se na konkrétní URL na webovém serveru, kde příslušný PHP skript tato data zpracovává.

3.4 Kamerové snímkování

Kamerové snímkování může sloužit pro historický záznam vizuální podoby počasí nebo jako informační zdroj o pohybu osob. Je realizováno originální Raspberry Pi kamerou.



Obrázek 12 – Originální Raspberry Pi kamera (GROBOTRONICS 2017)

Snímek je vyfocen, uložen v paměti Raspberry Pi, následně je pomocí FTP (File Transfer Protocol) nahrán na webový server a poté smazán z paměti mikropočítače. Snímky jsou číslovány a pojmenovány dynamicky na základě počtu již existujících snímků. V součinnosti s odesláním samotného snímku jsou odeslána i data o snímku obsahujícím jméno fotky a datum pořízení.

Automatické snímkování má vzorkovací periodu 5 minut, ale je jednoduchá možnost pro změnu.

Dalším použitím kamery je senzorové snímkování. Kamera pořídí snímek na základě vstupu z okolí. V této práci jsem využil už použitý magnetický senzor. Tato aplikace může být použita pro informace o pohybu osob v budově nebo na pozemku v případě vstupu přes snímané dveře nebo bránu. Snímkování je nastaveno tak, aby byl pořízen snímek pouze při otevření nikoliv při jakékoliv změně.

3.5 Ovládání světel

Ovládání světel nebo i jiných dvoupolohových zařízení je na Raspberry realizováno docela jednoduše. Důležité je mít nastavené příslušné piny jako výstupní.

```
if data[x]["deviceName"]=="Lamp" :
    if data[x]["state"] == "true":
        GPIO.setup(33, GPIO.OUT)
        GPIO.output(33, GPIO.HIGH)
    else:
        GPIO.setup(33, GPIO.IN)

if data[x]["deviceName"]=="LED1" :
    if data[x]["state"] == "true":
        GPIO.output(37, GPIO.HIGH)
    else:
        GPIO.output(37, GPIO.LOW)

if data[x]["deviceName"]=="LED2" :
    if data[x]["state"] == "true":
        GPIO.output(11, GPIO.HIGH)
    else:
        GPIO.output(11, GPIO.LOW)
```

Co se týče elektronického zapojení, musíme si uvědomit, že Raspberry nedokáže poskytovat dostatek elektrické energie pro domácí spotřebiče. Z toho důvodu musí Raspberry ovládat relé, které následně spíná samotný spotřebič. Vzhledem k maximálnímu přípustnému proudu na daném relé musíme znát výkon spotřebiče. Například u topení bychom zřejmě potřebovali relé, které ovládá ještě další relé. Protože topení může mít výkon v řádech kilo wattů. V naší aplikaci má žárovka výkon 40 W, a proto nám stačí použít jedno relé.

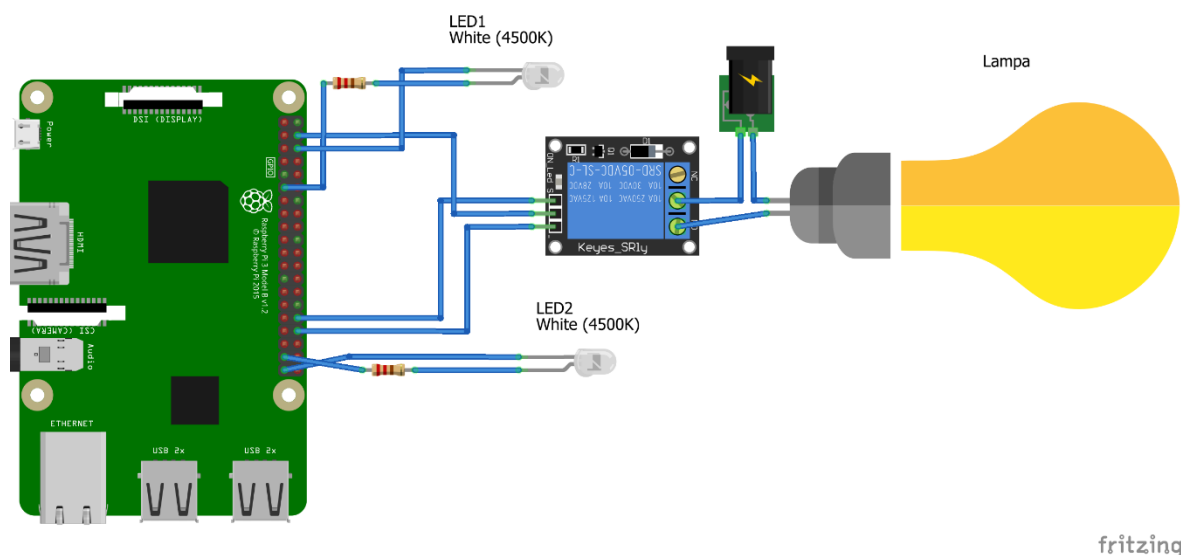
V úloze byl použit modul obsahující relé značky SONGLE, typ SRD-05VDC-SL-C pro proudy do 10 A.

Proud procházející žárovkou:

$$P = U * I$$

$$40 = 230 * I$$

$$I = 0,174 \text{ A}$$



Obrázek 13 – Schéma zapojení pro ovládání LED a lampy

3.6 Aplikace pro dotykový displej

Pro oficiální 7palcový dotykový displej byla vytvořena samostatná dotyková aplikace s využitím Kivy frameworku. Aplikace umožňuje ovládat světla, zobrazuje aktuální meteorologické údaje a zobrazuje aktuální obraz z kamery.



Obrázek 14 – Ukázka vzhledu aplikace pro dotykový displej

4 Webová aplikace

Webová aplikace je hostována na externím placeném serveru, a proto je přístupná odkudkoliv z globální sítě Internet. Základ je vytvořen v programovacím jazyce PHP a vizuální stránka pomocí HTML, CSS a Javascript. Pro ukládání dat je využita relační databáze MySQL.

PHP

Je to multiplatformní programovací jazyk pracující na straně serveru určený pro dynamické programování webových stránek. PHP má 82% podíl mezi programovacími jazyky pro webové stránky, které jsou na serverové části (duben 2017) a má velkou podporu u poskytovatelů hostingových služeb. PHP skript se provádí v pozadí a je možné vidět pouze jeho výstupy, nikoliv samotný kód. Datový typ proměnné je vázán na hodnotu a nikoliv na samotnou proměnnou, to se nazývá dynamické typování. Kód se píše vždy mezi značku `<?php` a `?>`. Poslední verze je 7.1.4 (květen 2017). (TVORBA-WEBU.CZ 2008) (W3TECHS 2017)

HTML

Je to značkovací jazyk pro tvorbu webových aplikací. Je založen na použití značek, které jsou umísťovány do ostrých závorek `<>`. Na rozdíl o PHP je viditelný pro uživatele.

Základ HTML dokumentu:

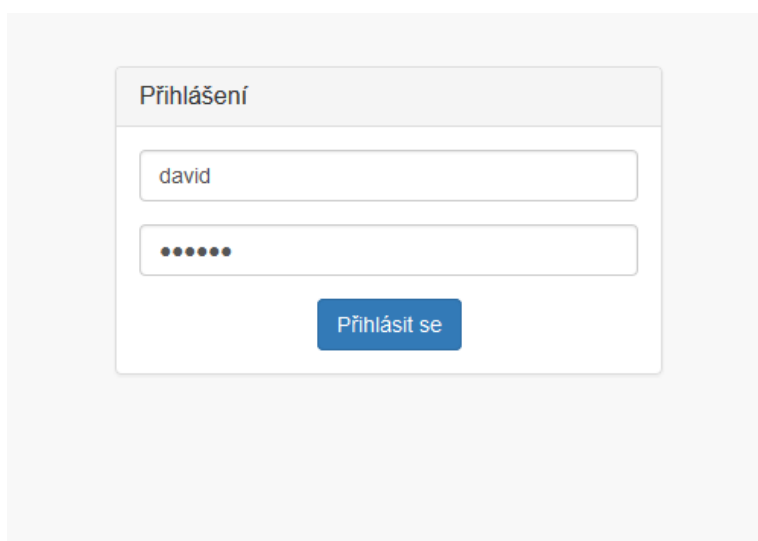
```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
  </body>
</html>
```

Javascript

Programovací jazyk s matoucím názvem, který má pouze část podobné syntaxe z jazyka Java, ale jinou filosofii. Javascript běží na straně klienta a umožňuje dynamickou změnu webové stránky, aniž by došlo ke komunikaci se serverem. Jasnou výhodou je zrychlení samotné aplikace v těch případech, že nepotřebujeme komunikovat se serverem. Javascript je interpretovaný programovací jazyk, což znamená, že je překládán v průběhu vykonávaného kódu. (ČÁPKA 2013)

Design webové aplikace je vytvořen pomocí bezplatné open-source (volně dostupné) šablony s názvem SB Admin 2, využívající sadu nástrojů Bootstrap. Grafy jsou řešeny pomocí nástroje Morris. (START BOOTSTRAP 2014)

Webová aplikace slouží jako hlavní přístupový bod pro uživatele a obsahuje všechny funkce. Pro zlepšení bezpečnosti aplikace vyžaduje přihlášení pomocí jména a hesla. Heslo je následně jednosměrně šifrováno a porovnáváno se stejně zašifrovaným heslem v databázi. Toto základní opatření je vhodné, pokud dojde k odcizení databáze. Ani admin s přístupem do databáze nemůže znát hesla svých uživatelů, protože nejsou uloženy v jejich originální formě.

The image shows a login form titled "Přihlášení" (Login). It consists of two input fields stacked vertically. The first field contains the text "david". The second field contains masked characters, represented by a series of dots. Below these fields is a blue button with the text "Přihlásit se" (Login).

Obrázek 15 – Přihlašovací formulář

Komunikace s Raspberry Pi využívá technologie Wi-Fi, což je označení pro standardy popisující bezdrátovou komunikaci a samozřejmě TCP/IP protokoly, které jsou hlavními protokoly pro síť Internet.

Systém řízení báze dat

V této práci je použita databáze MySQL, jde o tzv. relační databáze, což znamená, že je založena na tabulkách, kdy každá tabulka obsahuje položky jednoho typu. Položky jsou řazeny v řádcích. Sloupce jsou atributy. Každý sloupec má přesně definovaný datový typ.

Tabulka 1 – Struktura tabulky pro archivaci tlaku

Atribut	Typ	Délka	Klíč	Nulový	Výchozí
ID	INT	11	Primární	Ne	–
TLAK	INT	11	–	Ne	–
DATETIME	TIMESTAMP	–	–	Ne	CURRENT_TIMESTAMP

Příjem dat

Pro příjem dat slouží PHP skript očekávající data odeslaná metodou POST. Pro práci s MySQL databází byla v souladu s principem objektově orientovaného programování vytvořena třída, na které je poté volán objekt. Konstruktor třídy obsahuje přístupové údaje pro databázi a vytvoří spojení. Třída obsahuje mnoho metod, které vykonávají příkazy pro databázi, například vytvoření, zobrazení, smazání nebo upravení záznamu v tabulce. Skript na základě obsahu dat rozhodne, kterou metodu vykoná.

4.1 Ovládání světel

Cílem této úlohy je, aby byla možnost ovládat dvoupolohová zařízení odkudkoliv s přístupem k Internetu. V aplikaci jsou použity LED diody a lampa, ale použití může být i jiné než pouze ovládání světel, například topení, větráky, čerpadla aj.

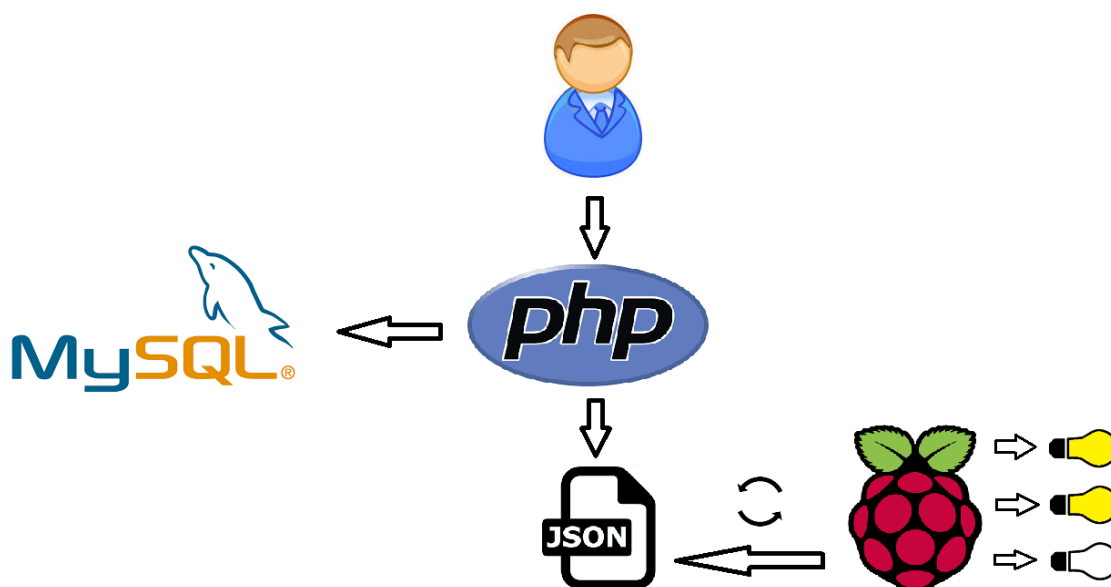
Data jsou v tomto případě odesílána z webového serveru na Raspberry Pi. Toto je realizováno nepřímo pomocí změny textového souboru, který si Raspberry cyklicky čte a na základě toho spíná jednotlivá zařízení. Data v tomto textovém souboru jsou uložena ve formátu JSON.

JSON

Zkratka pro JavaScript Object Notation. Je to odlehčený formát pro výměnu dat. Je snadno analyzovatelný a generovatelný strojově, ale i lehce zapisovatelný a čitelný pro programátora. Lze považovat za alternativu k XML. JSON je textový na programovém jazyce nezávislý formát. Data jsou zapisována v párech název/hodnota v objektu. Více záznamů může být obsaženo v poli. Můžeme pomocí něj přenášet jakékoliv datové struktury. Mnoho webových služeb pro své API používá právě JSON.

Ukázka JSON kódu:

```
[{"deviceName": "Lamp", "state": "true"},  
{"deviceName": "LED1", "state": "false"},  
{"deviceName": "LED2", "state": "false"}]
```

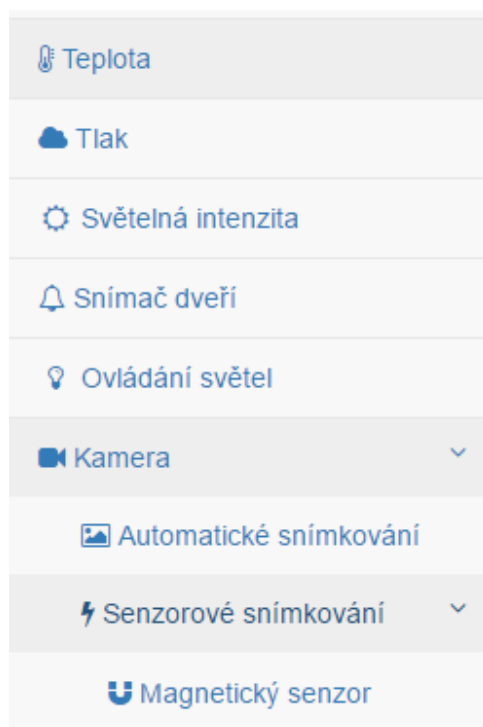


Obrázek 16 – Komunikační ovládací schéma

4.2 Vizualizace

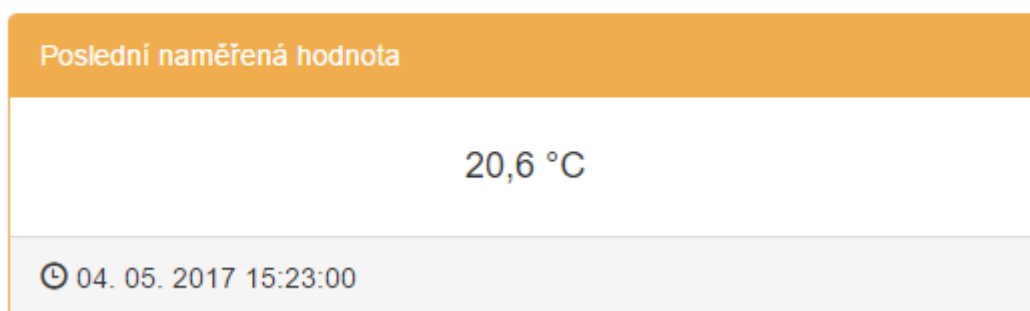
Důvod vzniku webové aplikace je hlavně vizualizace dat. Cílem je maximální přehlednost a jednoduchost pro uživatele. Díky responzivním vlastnostem nástrojů Bootstrap je web velmi robustní s ohledem na použité zařízení a hlavně rozlišení obrazovky zachovává správné poměry velikosti a čitelnosti.

Hlavní menu je umístěno na levé straně. Jedná se o víceúrovňové menu, které realizuje navigaci v aplikaci. Každá položka menu je doplněna o symbolickou ikonu pro zlepšení orientace uživatele.



Obrázek 17 – Levé menu

Teplota



Obrázek 18 – Zobrazení poslední naměřené hodnoty

Informační tabulka zobrazující poslední naměřenou hodnotu, dominuje každé stránce s meteorologickými údaji. Datum muselo být správně naformátováno, protože je v databázi uloženo jinak. To samé platí o číselných hodnotách obsahujících desetinná čísla. Ta jsou programově zpracována s tečkou a musí být přeformátována pro potřeby vizualizace podle české normy.

Metoda ve třídě *Database*, která navrátí poslední naměřenou hodnotu:

```
public function vypisPosledni($tabulka,$sloupec){  
    $sql = "SELECT " . $sloupec . ",DATEC FROM " . $tabulka  
    . " ORDER BY DATEC DESC LIMIT 1";  
  
    $result = $this->conn->query($sql);  
    while($row = $result->fetch_assoc()) {  
        $posledni[0] = $row[$sloupec];  
        $posledni[1] = $row["DATEC"];}  
  
    return $posledni;}
```

Vzhledem k tomu, že jsou meteorologické hodnoty vzorkovány s minutovou vzorkovací periodou, má uživatel možnost filtrovat data. Může se tedy rozhodnout pro podrobnější záznamy nebo méně podrobné. Dále formulář obsahuje dynamicky generované dny, kdy probíhalo měření a umožní uživateli vybrat si zvolený den.

Historie

Den 25. 04. 2017 ▾ Rozestup ▾

Graf

22 °

21 °

15. 04. 2017

16. 04. 2017

17. 04. 2017

18. 04. 2017

19. 04. 2017

20. 04. 2017

23. 04. 2017

24. 04. 2017

25. 04. 2017

29. 04. 2017

30. 04. 2017

04. 05. 2017

1 min

5 min

10 min

30 min

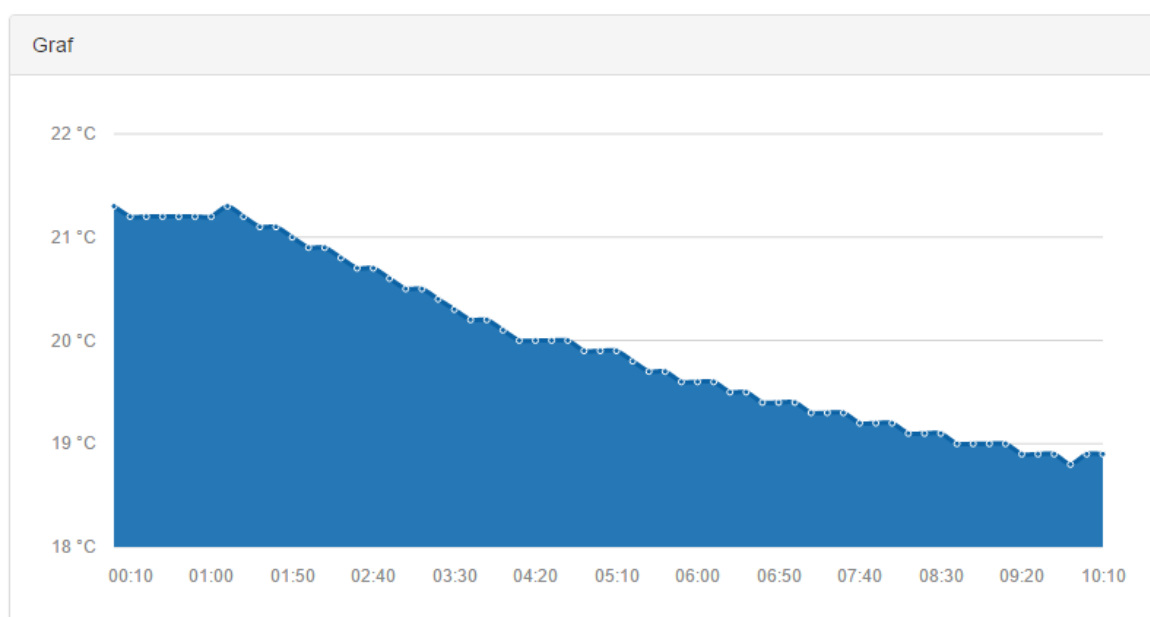
Obrázek 19 – Možnost volby zobrazení dat

Ukázka kódu pro možnost volby data a rozestupu:

```
<div class="form-group">
  <label for="select">Den</label>
  <select class="form-control" name="select_dny">
    <?php for ($x = 0; $x < sizeof($dny); $x++) {
      $dny_format = new DateTime($dny[$x]);
      $dny_format_done = $dny_format->format('d. m.
Y');

      echo "<option value='" . $dny[$x] . "'";
      if($dny[$x]==$_SESSION["vybrany_den"]){
        echo "selected";
      }
      echo">" . $dny_format_done . "</option>";
    }?>
  <div class="form-group">
    <label for="select">Rozestup</label>
    <select class="form-control"
name="select_rozestup"
onchange="this.form.submit()">
      <option value="" selected></option>
      <option value="1">1 min</option>
      <option value="5">5 min</option>
      <option value="10">10 min</option>
      <option value="30">30 min</option>
    </select>
  </div>
```

Grafické zobrazení historického trendu je realizováno pomocí nástroje Morris. Pro lepší přehlednost grafu je popis vertikální osy dynamicky generován podle naměřeného minima nebo maxima.



Obrázek 20 – Grafické zobrazení dat

Na obrázku 21 vidíme zobrazení hodnot pouze po deseti minutách, což odpovídá zadanému rozpětí. Vzhledem k možnosti velkého počtu řádků byl navíc přidán prvek rolovací lišty.

Tabulka	
Datum a čas	Teplota [°C]
25. 04. 2017 00:00:00	21,3
25. 04. 2017 00:10:00	21,2
25. 04. 2017 00:20:00	21,2
25. 04. 2017 00:30:00	21,2
25. 04. 2017 00:40:00	21,2
25. 04. 2017 00:50:00	21,2
25. 04. 2017 01:00:00	21,2
25. 04. 2017 01:10:00	21,3
25. 04. 2017 01:20:00	21,2

Obrázek 21 – Tabulkové zobrazení

Vizualizace extrémních hodnot v rámci vybraného dne využívá červené barvy pro maximální hodnotu a modré barvy pro minimální hodnotu. Rovněž je v dolní části uveden čas naměření.

Minimum
18,8 °C
09:50:00

Maximum
21,3 °C
00:00:00

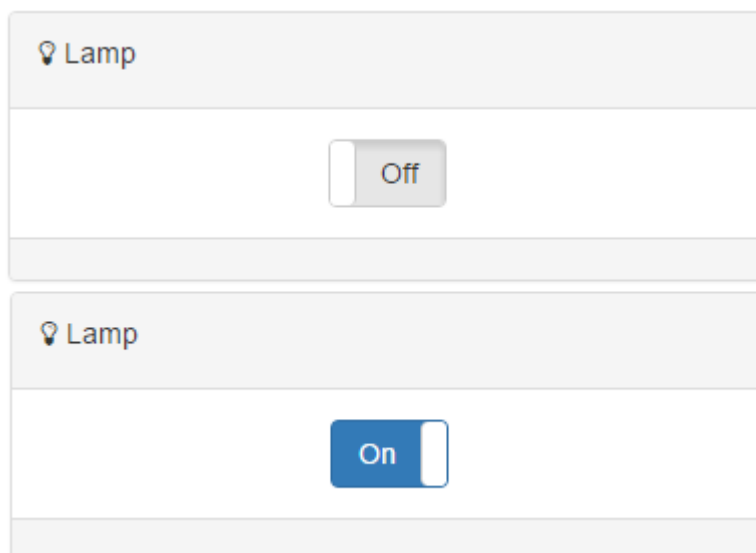
Obrázek 22 – Vizualizace extrémních hodnot

Metody pro zjištění extrémních hodnot v rámci vybraného dne:

```
public function vypisMin($tabulka,$sloupec,$datum){
    $sql = "SELECT " . $sloupec . ",DATEC FROM " . $tabulka .
    " WHERE date(DATEC) = '" . $datum . "' ORDER BY " . $sloupec . "
    ASC LIMIT 1";
    $result = $this->conn->query($sql);
    while($row = $result->fetch_assoc()) {
        $min[0] = $row[$sloupec];
        $min[1] = $row["DATEC"];
    }
    return $min;
}

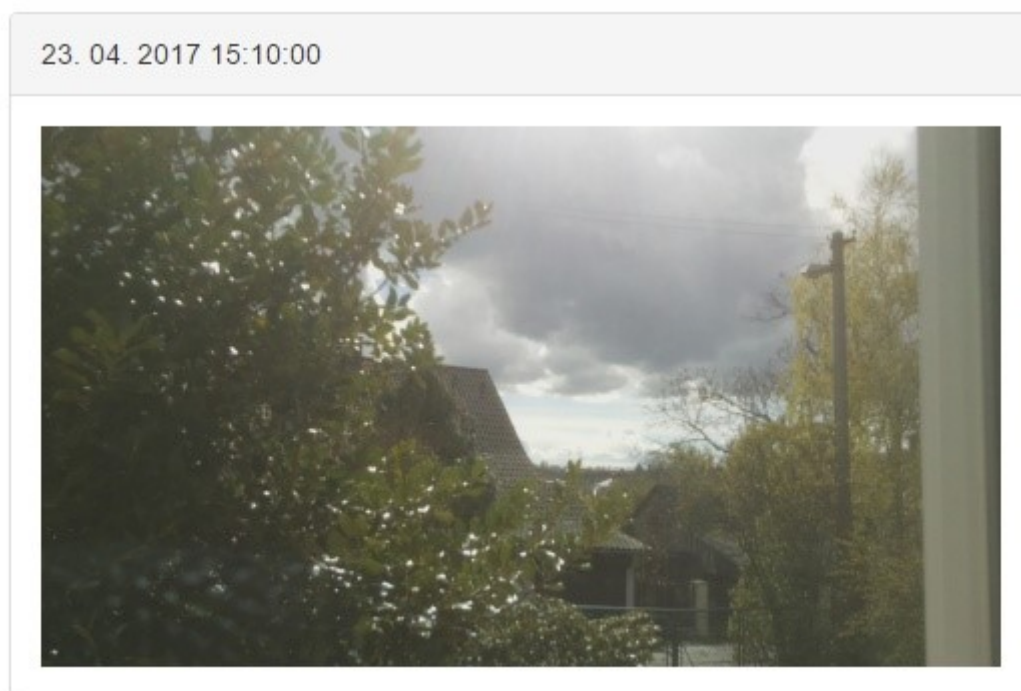
public function vypisMax($tabulka,$sloupec,$datum){
    $sql = "SELECT " . $sloupec . ",DATEC FROM " . $tabulka .
    " WHERE date(DATEC) = '" . $datum . "' ORDER BY " . $sloupec . "
    DESC LIMIT 1";
    $result = $this->conn->query($sql);
    while($row = $result->fetch_assoc()) {
        $max[0] = $row[$sloupec];
        $max[1] = $row["DATEC"];
    }
    return $max;
}
```

Ovládání světel je realizováno pomocí dvoustavového tlačítka tzv. Toggle Button. Díky tomu, že Raspberry odesílá data o aktuálním stavu světel, může se vypínač přepnout do odpovídající polohy sám při aktualizaci stránky. To se může stát, pokud někdo ovládá světla pomocí mobilní aplikace. Proto bude uživatel, který není v blízkosti světla, informován, jestli svítí nebo ne.



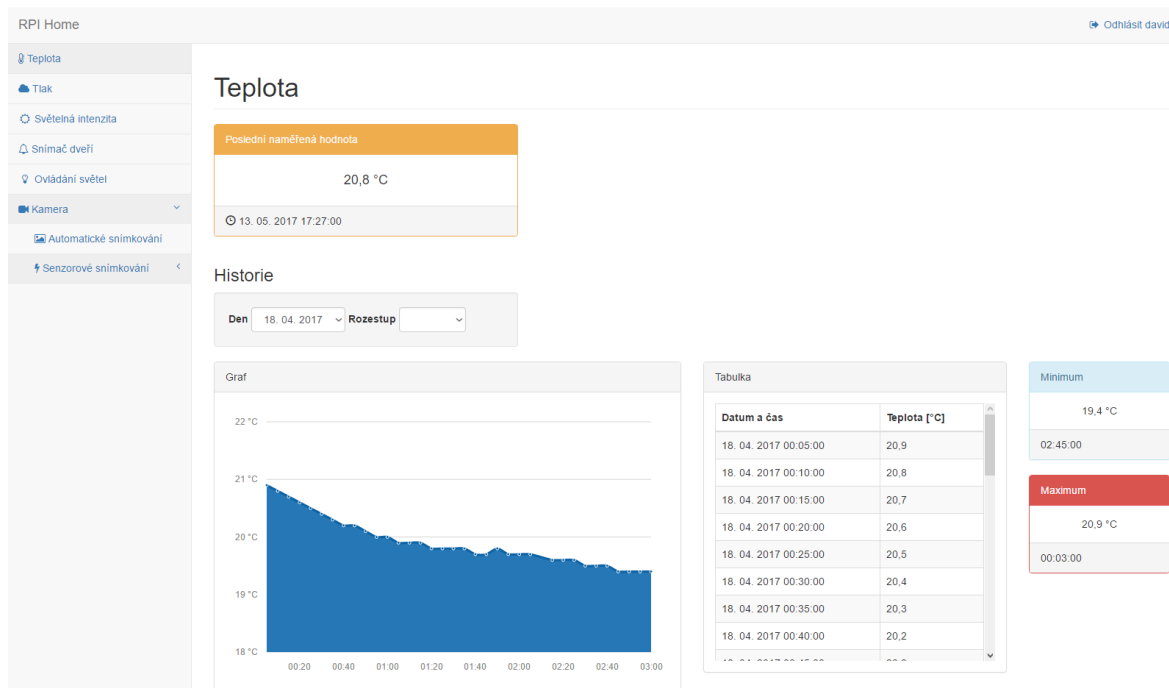
Obrázek 23 – Ovládání světel

U automatického snímání jsou obrázky dynamicky vykreslovány do třech sloupců. V horní části panelu je zobrazeno datum pořízení snímku. Snímky jsou při této vizualizaci zmenšeny na velikost daného panelu. Ale je možné na ně kliknout a následně je uživatel přesměrován přímo na zdrojový obrázek v plné velikosti.



Obrázek 24 – Zobrazení automaticky vyfoceného snímku

V pravém horním rohu aplikace se nachází tlačítko pro odhlášení, které vymaže aktuální informace o přihlášení a přesměruje uživatele na přihlašovací stránku. Dále je také v každém skriptu ověřováno, zda je uživatel přihlášen, pokud ne, dojde rovněž k přesměrování na přihlašovací stránku. Je to zároveň bezpečnostní prvek pro případ, pokud by někdo chtěl obejít přihlášení zadáním konkrétní adresy.



Obrázek 25 – Celkový pohled

5 Mobilní aplikace

Pro tuto práci byla vytvořena nativní aplikace pro zařízení s operačním systémem Android. Výhodou nativních aplikací je to, že plně dokážou využít schopností telefonu. Nativní aplikaci musíme nainstalovat do svého zařízení, to můžeme brát jak pozitivně tak negativně. Pozitivní však je, že aplikace může fungovat i v offline režimu na rozdíl od webové aplikace. Nativní aplikace je rychlejší než ostatní možnosti vývoje mobilních aplikací, je to způsobeno právě tím, že aplikace běží přímo na daném zařízení. Co se týká vývoje aplikace, je snadnější vyvinout nativní aplikaci, protože používá robustní programovací jazyky (Java, C++, atd.), které mají za sebou bohatou historii. Nativní aplikace se musí potýkat s různými verzemi operačních systémů a také s rozlišením.

Ve své práci jsem se zabýval vývojem mobilní aplikace pro operační systém Android, neboť má největší zastoupení na trhu s chytrými telefony jak v České republice, tak i ve světě.

Vývoj aplikace probíhal v Android Studiu ve verzi 2.3.1. Aplikace je vytvořena pro telefony s API Android 14 a vyšší neboli verze 4.0 a vyšší. Primární testování probíhalo na mobilním telefonu s verzí 4.3. Grafické prvky jsou napsány v XML souboru. Pro tuto aplikaci byly použity různé druhy *Layoutů*, dále prvek *Button* a *TextView*.

Část XML souboru pro vytvoření tlačítka (*Button*):

```
<Button
    android:id="@+id/btnled1off"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_alignParentBottom="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_weight="1"
    android:text="LED 1&#10;OFF"
    android:textSize="18sp" />
```

XML kód pro zobrazení textu (*TextView*):

```
<TextView
    android:id="@+id/sensorView0"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/sensorView1"
    android:layout_centerHorizontal="false"

    android:layout_gravity="center_vertical|center_horizontal"
    android:layout_marginTop="20dp"
    android:layout_weight="1"
    android:text="Teplota: "

    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textSize="24sp" />
```



Obrázek 26 – Mobilní aplikace

Mobilní aplikace komunikuje s Raspberry pomocí bezdrátové technologie Bluetooth. Bluetooth je technologie pro bezdrátový přenos dat mezi dvěma zařízeními. Teoretická

rychlost této technologie je 2,1 Mbps a rozsah 100 m. Aby vůbec mohla komunikace proběhnout, obě zařízení musí být propojena (synchronizována).

Najednou můžete s vaším zařízením spojit až 8 dalších. Je pomalejší než Wi-Fi, ale je snadněji nastavitelná a méně náročná na spotřebu baterie.

Raspberry Pi slouží jako server. Je využíván RFCOMM protokol. V Python skriptu je definován tzv. UUID, které musí být vytvořeno pro každý server, mělo by být originální. Po spuštění skriptu se čeká na připojení klienta a následně může probíhat komunikace. Ve vytvořené mobilní aplikaci je možnost ovládat světla a zobrazit aktuální hodnoty meteorologických dat, které jsou aktualizovány každých 5 sekund. Data z Raspberry jsou odesílána v jednom řetězci, který obsahuje úvodní znak #, poté oddělovač veličin * a ukončovací znak ~.

Kód mobilní aplikace pro zpracování přijatého řetězce:

```
String readMessage = (String) msg.obj;
recDataString.append(readMessage);
int endOfLineIndex = recDataString.indexOf("~");
if (endOfLineIndex > 0) {
    String dataInPrint = recDataString.substring(0,
        endOfLineIndex);
    int dataLength = dataInPrint.length();
    ArrayList<String> hodnoty;
    hodnoty = new ArrayList<>();
    int kon = 1;
    if (recDataString.charAt(0) == '#'){
        for(int i = 1; i <= dataLength; i++){
            if(recDataString.charAt(i) == '*'){
                hodnoty.add(recDataString.substring(kon,
                    i));
                kon = i+1;}
        }
    }
    sensorView0.setText("Teplota = " + hodnoty.get(0) + " °C");
    sensorView1.setText("Tlak = " + hodnoty.get(1) + " Pa");
    sensorView2.setText("Světelná intenzita = "+
        hodnoty.get(2)+"lx");
}
```

6 Závěr

Jedním z cílů práce bylo popsat platformu Raspberry Pi, její historii a technické parametry. Jedná o mikropočítač za zajímavou nákupní cenu, který díky svým vstupům a výstupům může realizovat akční zásahy nebo obstarávat měření. Nejnovější aktuální model je Raspberry Pi 3 Model B, který byl použit pro tuto práci. Model obsahuje 40 vstupů/výstupů s různými specifikacemi, například piny s napětím 5 V, 3,3 V, uzemnění nebo piny pro I2C sběrnici. Jeho výhodou oproti předchozím modelům jsou integrované technologie Wi-Fi a Bluetooth. Jako operační systém pro tuto práci byl zvolen Raspbian, který obsahuje předinstalované vývoje prostředí. Skripty, které jsou spouštěny na Raspberry Pi, jsou napsány v jazyce Python.

Práce obsahuje aplikace pro měření teploty, tlaku, světelné intenzity, detekce uzavření/otevření dveří, ovládání světel, automatické a senzorové snímkování. Pro měření tlaku byl použit senzor BMP180, který lze použít i pro měření teploty. Pro měření světelné intenzity byl zvolen senzor TSL2561. Oba senzory komunikují pomocí I2C sběrnice. Kamerové snímkování obstarává oficiální kamerový model pro Raspberry Pi. Pro detekci uzavření/otevření dveří byl využit jednoduchý magnetický senzor. Dále byla vyvinuta aplikace pro sedmipalcový dotykový displej pomocí Kivy frameworku. Aplikace umožňuje zobrazovat aktuální meteorologická data, ovládat světla a zobrazuje aktuální náhled kamery. Součástí práce jsou schémata zapojení jednotlivých obvodů, které byly vytvořeny v programu fritzing.

Naměřená data se průběžně s danou vzorkovací periodou odesílají na webový server. Pro meteorologická data je vzorkovací perioda 1 minuta a pro automatické snímky 5 minut. Data z magnetického senzoru jsou odesílána okamžitě. Komunikace je řešena pomocí TCP/IP protokolu a technologie Wi-Fi. Data jsou odesílána pomocí požadavku POST, který je na webovém serveru zpracován pomocí PHP skriptu a následně dojde k uložení dat do MySQL databáze. Pro přístup a práci s databází byla vytvořena třída obsahující mnoho metod, například výpis maximální hodnoty v rámci dne nebo výpis poslední hodnoty atd. Pro ovládání světel je komunikace opačná, ale vzhledem k nemožnosti vyslat požadavek přímo na Raspberry Pi, webový server zapisuje data do textového souboru ve JSON formátu, který Raspberry Pi neustále cyklicky vyhodnocuje.

Pozadí (backend) webové aplikace je napsáno v PHP, které pracuje na straně serveru a je možné vidět pouze jeho výstupy, nikoliv samotný kód. Vizualní uživatelská část je realizována pomocí HTML, CSS a Javascript.

Design webové aplikace je vytvořen pomocí bezplatné open-source (volně dostupné) šablony s názvem SB Admin 2 využívající sadu nástrojů Bootstrap. Grafy jsou řešeny pomocí nástroje Morris. Cílem vizualizace je maximální přehlednost a jednoduchost pro uživatele. Vizualizace obsahuje historické trendy v grafech, tabulky, extrémní hodnoty v daný den. Uživatel má rovněž možnost výběru příslušného dne a volí měřítko, s jakým na data bude nahlížet a jak se mu budou zobrazovat.

Dále byla v rámci práce vytvořena mobilní aplikace pro zařízení s operačním systémem Android, která komunikuje pomocí Bluetooth technologie. I v tomto případě je komunikace obousměrná. Je využíván RFCOMM protokol. Raspberry je server a mobilní aplikace k němu přistupuje jako klient. Hlavní cílem mobilní aplikace je jednoduchost. Cílem je vytvořit pouze jakýsi ovladač pro světla a jiné dvoustavové spotřebiče a zobrazení aktuálních hodnot meteorologických snímačů.

Hlavním cílem práce bylo vytvořit nástroje pro využití v domácnostech, kde bude jako ústřední prvek působit mikropočítač Raspberry Pi a dále periferní aplikace pro pohodlné a přehledné zobrazení dat. Práce je zaměřena hlavně na programování.

Možný směr dalšího řešení je použití více senzorů, bezdrátová komunikace se senzory, možnost regulace nebo webová aplikace pro místní síť.

Seznam použité literatury

ALLEN, Grant. *Android 4: průvodce programováním mobilních aplikací*. Brno: Computer Press, 2013. ISBN 9788025137826.

Android Developer portal . Dostupné z: <https://developer.android.com/index.html>

BENCHOFF, Brian. INTRODUCING THE RASPBERRY PI 3. In: HACKDAY [online]. 2016 [cit. 2017-05-05]. Dostupné z: <http://hackaday.com/2016/02/28/introducing-the-raspberry-pi-3/>

BOSCH. *BMP180 Digital pressure sensor* [online]. 2013 [cit. 2017-05-05]. Dostupné z: <https://cdn-shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf>

CROSTON, Ben. *Raspberry GPIO Python* [online]. 2017 [cit. 2017-05-05]. Dostupné z: <https://sourceforge.net/p/raspberry-gpio-python/wiki/Inputs/>

ČÁPKA, David. *1. díl - Úvod do JavaScriptu* [online]. 2013 [cit. 2017-05-05]. Dostupné z: <https://www.itnetwork.cz/javascript/zaklady/javascript-tutorial-uvod-do-javascriptu-nepochopeny-jazyk>

ENGADGET. Raspberry Pi A/B. In: Engadget [online]. 2016 [cit. 2016-05-15]. Dostupné z: <http://www.engadget.com/products/raspberry-pi/a-b/specs/>

EMBEDDED LAB. *BMP180 sensor module*. [online] Dostupné z: <http://embedded-lab.com/blog/wp-content/uploads/2015/01/BMP180.jpg>

GE TECH WIKI. Raspberry Pi Model B+. In: GE tech wiki [online]. 2014 [cit. 2016-05-15]. Dostupné z: [http://www.geeetech.com/wiki/index.php/Raspberry_Pi_Model_B%](http://www.geeetech.com/wiki/index.php/Raspberry_Pi_Model_B%2B#Specifications)

[2B#Specifications](http://www.geeetech.com/wiki/index.php/Raspberry_Pi_Model_B%2B#Specifications)

GROBOTRONICS. *Raspberry Pi Camera Module*. [online] Dostupné z: https://grobotronics.com/images/detailed/14/DSC_0393-800x609.jpg

GXC11. *1. díl - Úvod do Pythonu* [online]. 2014 [cit. 2017-05-05]. Dostupné z: <https://www.itnetwork.cz/python/zaklady/python-tutorial-uvod-do-pythonu-a-zakladni-matematicke-operace>

HALFACREE, Gareth. *Raspberry Pi 3 Model B*. [online] Dostupné z: https://cdn.infoq.com/statics_s1_20160414-0116u1/resource/news/2016/02/raspberry-pi-3/en/resources/24914484549_c6281a9946_k.jpg

HORNÍK, Petr. *Pull-up rezistory* [online]. 2015 [cit. 2017-05-05]. Dostupné z: <http://www.elektroraj.cz/2015/12/16/pull-up-rezistory/>

JONES, David. About Windows 10 IoT. In: E101 [online]. 2015 [cit. 2016-05-15]. Dostupné z: <http://embedded101.com/Blogs/David-Jones/entryid/620/about-windows-10-iot>

JSON.ORG. *Úvod do JSON* [online]. 2016 [cit. 2017-01-19]. Dostupné z: <http://www.json.org/json-cz.html>

Kivy Pie *Kivy onRaspberry Pi - Python framework for multitouch apps*. Dostupné z: <http://kivypie.mitako.eu/kivy-pie.html>

MG SUPER LABS. *TSL2561 digital luminosity / lux / light sensor*. [online] Dostupné z: https://www.mgsuperlabs.co.in/estore/image/cache/data/ADAFRUIT/439/MFG_439-500x500.jpg

MICROSOFT. *Raspberry Pi 2 & 3 Pin Mappings*. [online] Dostupné z: https://az835927.vo.msecnd.net/sites/iot/Resources/images/PinMappings/RP2_Pinout.png

NIELD, David. *What is Bluetooth?* [online]. 2016 [cit. 2017-01-18]. Dostupné z: <http://www.techradar.com/how-to/computing/what-is-bluetooth-1323284>

Raspberry Pi Official web sites. Dostupné z: <https://www.raspberrypi.org/resources/learn/>

ROOT.CZ. Fotogalerie: Historie Raspberry Pi: od prototypu po verzi 3. In: ROOT.CZ [online]. 2016 [cit. 2016-05-15]. Dostupné z: <https://www.root.cz/galerie/historie-raspberry-pi-od-prototypu-po-verzi-3/#1>

SLEDGE. Co je Raspberry Pi? In: RASPI.cz :: Raspberry Pi [online]. 2011 [cit. 2016-05-15]. Dostupné z: <http://www.raspi.cz/2011/12/co-je-raspberry-pi/>

START BOOTSTRAP. *SB Admin 2* [online]. 2014 [cit. 2017-05-05]. Dostupné z: <https://startbootstrap.com/template-overviews/sb-admin-2/>

THE PI HUT. *Raspberry Pi Model A+ vs Model B+* [online] Dostupné z: https://cdn.shopify.com/s/files/1/0176/3274/files/model-b-vs-model-a_grande.jpg?7227

THE RASPBERRY PI FOUNDATION. Raspberry Pi FAQs. In: Raspberry Pi [online]. 2016 [cit. 2016-05-15]. Dostupné z: <https://www.raspberrypi.org/help/faqs/>

TVORBA-WEBU.CZ. *PHP základy* [online]. 2008 [cit. 2017-05-05]. Dostupné z: <https://www.tvorba-webu.cz/php/>

TVORBA-WEBU.CZ. *CSS styly* [online]. 2008 [cit. 2017-05-05]. Dostupné z: <https://www.tvorba-webu.cz/css/>

VYVOJ.HW.CZ. *Stručný popis sběrnice I2C* [online]. 2000 [cit. 2017-05-05]. Dostupné z: <http://vyvoj.hw.cz/navrh-obvodu/strucny-popis-sbernice-i2c-a-jeji-prakticke-vyuziti-k-pripojeni-externi-eeeprom-24lc256>

W3TECHS. *Historical trends in the usage of server-side programming languages for websites* [online]. 2017 [cit. 2017-05-05]. Dostupné z: https://w3techs.com/technologies/history_overview/programming_language

YOSHIDA, Steve. *Raspberry Pi Model A on the left, B model on the right.* [online] Dostupné z: <http://stevesprojectpages.com/wp-content/uploads/2013/08/>

RaspiModelAandB.jpg

Seznam použitých obrázků

Obrázek 1 – Raspberry Pi 3 Model B (HALFACREE 2017)	12
Obrázek 2 – Porovnání RPI Model A (vlevo) s RPI Model B (vpravo) (YOSHIDA 2013)..	14
Obrázek 3 – Porovnání RPI model A+ (nahore) a model B+ (dole) (THE PI HUT 2017)..	16
Obrázek 4 – Přehled pinů Raspberry Pi 3 Model B (MICROSOFT 2017)	17
Obrázek 5 – Senzor BMP180 (EMBEDDED LAB 2015)	19
Obrázek 6 – Grafické schéma zapojení senzoru BMP180	20
Obrázek 7 – Elektrické schéma zapojení senzoru BMP180.....	20
Obrázek 8 – Senzor TSL2561 (MG SUPER LABS 2017)	21
Obrázek 9 – Grafické schéma zapojení senzoru TSL2561	22
Obrázek 10 - Elektrické schéma zapojení senzoru TSL2561	22
Obrázek 11 – Grafické schéma zapojení magnetického senzoru	23
Obrázek 12 – Originální Raspberry Pi kamera (GROBOTRONICS 2017)	25
Obrázek 13 – Schéma zapojení pro ovládání LED a lampy	27
Obrázek 14 – Ukázka vzhledu aplikace pro dotykový displej.....	27
Obrázek 15 – Přihlašovací formulář	29
Obrázek 16 – Komunikační ovládací schéma.....	31
Obrázek 17 – Levé menu	32
Obrázek 18 – Zobrazení poslední naměřené hodnoty	32
Obrázek 19 – Možnost volby zobrazení dat.....	33
Obrázek 20 – Grafické zobrazení dat	34
Obrázek 21 – Tabulkové zobrazení.....	35
Obrázek 22 – Vizualizace extrémních hodnot.....	35

Obrázek 23 – Ovládání světel	37
Obrázek 24 – Zobrazení automaticky vyfoceného snímku.....	37
Obrázek 25 – Celkové pohled	38
Obrázek 26 – Mobilní aplikace	40